Available online at www.sciencedirect.com

ScienceDirect

Procedia Technology 17 (2014) 223 – 230

Procedia Technology

Conference on Electronics, Telecommunications and Computers – CETC 2013

Ensemble feature ranking applied to medical data

Vítor Santos[a], Nuno Datia[a], M.P.M. Pato[a,*]

[a]*ISEL, Rua Conselheiro Emídio Navarro 1, 1959-007 Lisbon, Portugal*

Abstract

Reduce the feature space in classification is a critical, although sensitive, task since it depends on a certain definition of *relevance*. Feature selection has been the motivation for many researchers. In medical datasets, relevant attributes are often unknown *a priori*. Feature selection provides the features that contribute most to the classification task *per si*, which should therefore be used by any classifier to produce a classification model. However, the dimension of the feature space may not allow the application of feature selection algorithms, due time and space complexity. In this work, we are concerned on the application of an efficient feature ranking algorithm for a given breast cancer dataset, that overcome the dimensionality of the data.

Peer-review under responsibility of ISEL – Instituto Superior de Engenharia de Lisboa, Lisbon, PORTUGAL.

Keywords: Breast Cancer; Data mining; Feature Ranking; Undersampling; SMOTE; R

1. Introduction

Currently, the motivation for applying feature selection (FS) techniques has been increasing, becoming a prerequisite for modelling most datasets [16,37]. In recent years, data has become increasingly larger in both number of instances and number of feature in many scientific disciplines such as medicine [1,2,8,14,34], engineering [15,19], customer relationship management [3,4,31], image retrieval [11,30,36] and others. This is a challenge that present and future research face, when apply knowledge discovery techniques to real–world problems.

High dimensional data can contain redundant or irrelevant information which may degrade the performance of learning algorithms [26]. Ranking is a way to evaluate which features are relevant. Selected a minimal set of features, given a specific criteria, will produce simpler models, that take less time to compute and become more understandable. Furthermore, by requiring less resources, predict a sample becomes more affordable. Many machine learning algorithms are designed to learn which are the most appropriate attributes to use for making their decisions, that is, search in the space of attributes for the subset that is most likely to better predict the class. Essentially, when there are too many features in the problem, dimensionality reduction, through weak feature removal is highly desirable. Recent research has shown common machine learning algorithms to be adversely affected by irrelevant and redundant training information.

* Corresponding author. Tel.: +351-218-317-000 ; fax: +351-218-317-162.
E-mail address: mpato@deetc.isel.pt

Peer-review under responsibility of ISEL – Instituto Superior de Engenharia de Lisboa, Lisbon, PORTUGAL.
doi:10.1016/j.protcy.2014.10.232

The number of training examples needed to reach a given accuracy level grows exponentially with the number of irrelevant attributes [20,23,24]. Sample complexity for decision tree algorithms can grow exponentially on some concepts (such as parity) as well. The Naïve Bayes classifier can be adversely affected by redundant attributes due to its assumption that attributes are independent given the class [23]. Decision tree algorithms such as C4.5 [28,29] can sometimes overfit training data, resulting in large trees. In many cases, removing irrelevant and redundant information can result in C4.5 producing smaller trees [21]. Experiments show that feature ranking using linear Support Vector Machine (SVM) models yields good performance, even when training and testing data are not identically distributed [16,17]. For such reasons, it is usual to set FS as a fundamental data preparation process in any data mining tasks [16,37]. But what happens when a dataset is so large, that FS algorithms are impossible to apply, due to space and time complexity?

This work concerns on the application of FS techniques in such situations, namely, using feature ranking (FR) algorithms. In contrast to other dimensionality reduction techniques, like those based on projection (e.g. principal component analysis) or compression (e.g. using information theory), FS techniques do not alter the original representation of the variables, but merely select a subset of them. Thus, they preserve the original semantics of the variables, hence, offering the advantage of interpretability by a domain expert.

In the next section, we first review models of FS to reduce the dataset and explain why a filter solution is suitable for high dimension feature spaces. Section 3 describes our feature ranking approach. In section 4, we evaluate the efficiency and effectiveness of this algorithm with other representative FS algorithms, and discuss the implications of the findings. Finally, section 5 discusses the merits and disadvantages of the various performances measures. Moreover, we conclude with some possible extensions.

2. Dimensionality reduction

In most learning algorithms, the complexity depends on the number of input features, d, as well as on the size of the data sample, N. Decreasing d also decreases the complexity of the inference algorithm during testing, as it holds out the possibility of more effective and rapid operation of data mining algorithms. In some cases, as a result of feature selection, accuracy on classification can be improved, with an easily interpreted model representation [18].

There are two main methods for reducing dimensionality of feature space: (i) feature selection, and (ii) feature extraction. The objective of FS is to identify some features in the dataset as important, and discard the remaining, unimportant dimensions [17,18]. The *best* subset, or the *novel* set, contains the least number of dimensions that most contributes to accuracy. On the other hand, the objective of feature extraction is finding a new set of k dimensions that are combinations of the original d dimensions [16]. Feature extraction and dimension reduction can be combined in one step using principal component analysis (PCA), linear discriminant analysis (LDA), or canonical correlation analysis (CCA) techniques as a preprocessing stepIn machine learning this process is also called low–dimensional embedding [5,35]. Figure 1 shows the systematics of dimension reduction arguments.

2.1. Feature selection

Feature selection algorithms fall into three broad categories, (i) the filter, (ii) the wrapper, or (iii) embedded [10,32]. Filter methods [9,38] select the best features according to some prior knowledge (commonly, feature evaluation metric score) and use the selected features instead of the error rate to score a feature subset. Filters select the features independently of the classifier and basically serve as a preprocessing step of feature pruning to ease the burden of classification. In general, filter methods are less computationally intensive, since they do not incorporate learning, but they produce a feature set which is not tuned to a specific type of predictive model. Many filters provide a feature ranking rather than an explicit best subset, and the cut off point in the ranking is chosen via cross-validation.

Wrapper methods [21], on the other hand, do not rely only on prior knowledge, but evaluate the feature subsets in a real classifier and evaluate their classification performance to select the features. Each novel subset is used to train a model, which is tested on a hold-out set. Wrapper methods use a search algorithm along with evaluation measures to find the optimal reduced feature set. Wrapper methods are very computationally intensive, since they typically need to run and evaluate the feature subsets in the classifier at every iteration. However, they provide the best performing feature set for that particular type of model.

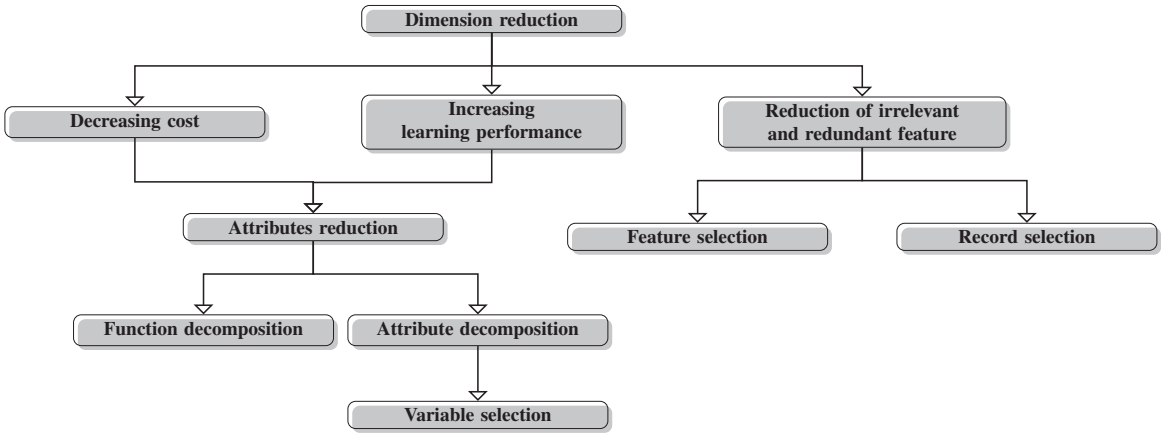

Fig. 1. Taxonomy for dimension reduction reasons.

In embedded methods [17,22], the search for the optimal feature subset is built into the classifier construction, and can be seen as a search in the combined space of feature subsets and hypotheses. Like wrapper approaches, embedded methods are also classifier specific. As the embedded methods incorporate FS in the training of the classifier and enable efficient algorithms to reach the optimum, these are faster than the wrapper methods. The example of this approach is the recursive feature elimination algorithm, commonly used with SVM to repeatedly construct a model and remove features with less weights. These approaches tend to be between filters and wrappers in terms of computational complexity.

As described above, on this work feature selection problem is a sub problem of dimension reduction. Feature ranking can be used in FS by first determining which available features are most influential for a classification task. In the feature ranking algorithm, a subset of features is often selected from the top of a ranking list. The process of selection involves two phases. The algorithm begins with a phase where features are individually evaluated, and provide a ranking according to a filter criterion. Afterwards, a feature evaluator is applied to a fixed number from the previous ranking, that is greater than a threshold value or the first k features ($k < d$).

3. Feature ranking approach

In this paper, we have developed an efficient FR algorithm for selecting the more relevant features prior to derivation of classification predictors. It uses a scoring function as ranking criterion to evaluate the correlation measure between each feature and the classes. This function comprises three measures for each class: the statistical between-class distance, the interclass overlapping measure, and an estimate of class impurity. In order to compute the statistical parameters used in these measures, a normalized form of histogram, obtained for each class, is employed as its a priori probability density. Since the proposed algorithm examines each feature individually, it provides a fast and cost-effective method for FR. We have tested the effectiveness of our approach on some benchmark data sets with high dimensions. For this purpose, some top-ranked features are selected and are used in some rule-based classifiers as the target data mining task.

The solution adopted was to run the same four FR algorithms several times with a random, small, subset in each iteration. The result of each run is a rank order for each feature, that is combined with the results from previous runs, acting as a ensemble FR, as shown in Figure 2. This technique is inspired in the Monte Carlo algorithm, which states that an outcome can be achieved by combining random successive approximations to the same result [12,27].

The ranks of each attribute in each iteration are combined, using a weight function, influenced by the position of the attribute in the partial rank. The result of the function gives a rank order for each feature, that is used to select the most important feature for determine the dependent variable, as illustrated in Algorithm 1. By definition, the last feature is the dependent variable.

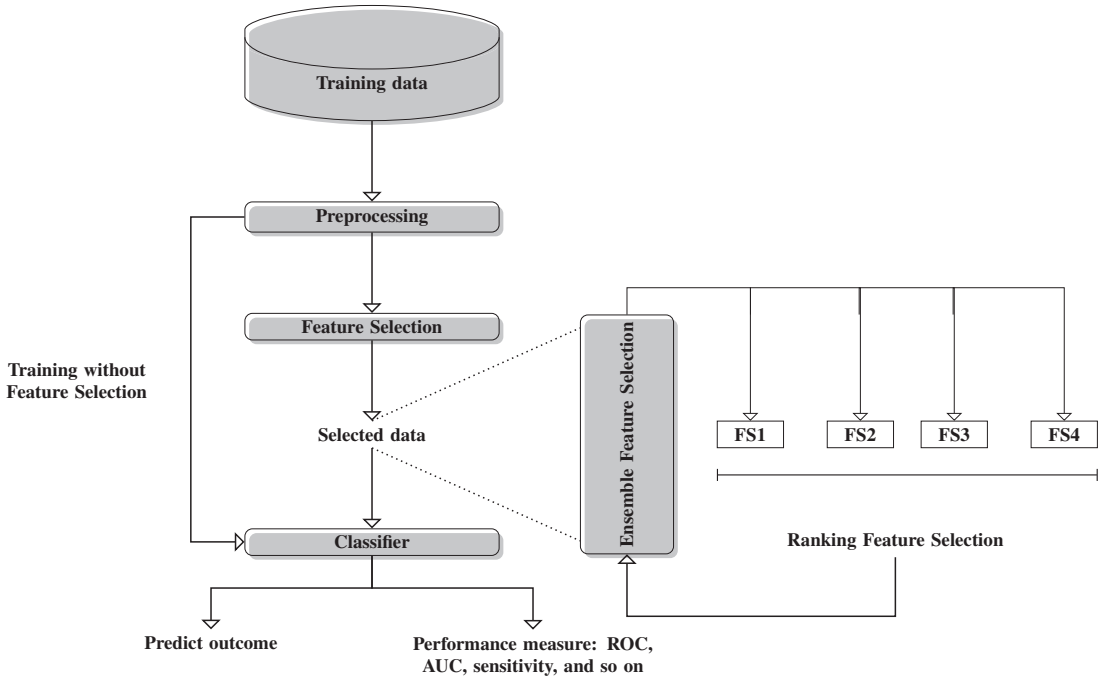

Fig. 2. Schematic process for the ensemble feature ranking.

Algorithm 1 Ensemble feature ranking

Input: *data* - The dataset for feature ranking, represented as a matrix;
N - The number of iterations;
sampleSize - The size of the sample for each iteration;
W - A vector containing the weights for each rank position;
algo - A vector of feature ranking functions
Output: A vector with the rank of each feature

EnsembleFeatureRanking(*data*, *N*, *sampleSize*, *W*, *algo*)

```
1  // A matrix of weights. Each row i represents the weights of the j features, given each algorithm in algo
2  weights = vector(algo.length)
3  for each fun in algo
4      localWeights = vector(N)
5      for n = 1 to N
6          randomData = SamplingWithReplacement(data, sampleSize)
7          localWeights.add(fun(randomData))
8      weights.add(calculateWeightedRank(localWeights, W))
9  rank = vector with the sum of all weights_j
10 return rank
```

calculateWeightedRank(*weights*, *W*)

```
1  weightedRank = vector(W.length)
2  set 0 in all weightedRank[i]
3  for i = 1 to N
4      weightedRank+ = W indexed by weights[i]
5  return weightedRank
```

3.1. Data analysis

The dataset used in this work is available at the KDD Cup 2008 website[1]. The challenge focuses the problem of early detection of breast cancer from X-ray images of the breast. We have settled six different datasets, from $[T_1..T_6]$,

[1] http://www.kdd.org/kdd-cup-2008-breast-cancer

as it is shown in Table 1. They include a combination of instance selection (IS) and FS, and are used to compare the classification performance with and without FR. In the case of FS, three approaches are used: (i) select all features, (ii) remove the redundant features based on correlation and, (iii) select the top 66 features with our FR approach.

Table 1. Dataset combination.

	Undersampling	SMOTE
Feature Ranking	T_1	T_4
Feature Selection	T_2	T_5
All Features	T_3	T_6

Two techniques are used to select instances, such as, random undersampling [13] and SMOTE [7]. In the first case, we produce datasets with 1246 instances, which contains all positive cases and 623 randomly negative instances. In the second one, the datasets have 2000 instances, with approximate class distribution.

The learning algorithms used in this experimental comparison are: the Support Vector Machines (SVM), Bagging using the RPART function (BAG), Random Forest (RF) and Naïve Bayes (NB). These algorithms are available in R[2] environment. The classifiers are compared using: Area Under the Curve (AUC), sensitivity (Patient), and false positive (FPR) rate (instance) based on work of Santos [33].

4. Results and discussion

The objective of this section is to evaluate our proposed algorithm in terms of number selected features, and learning accuracy on selected features. Feature selection requires metrics for evaluating the importance of the individual features. Several evaluation metrics have been used in different tasks. We use three criteria to compare the classifiers: AUC, sensitivity, and FPR. The best classifier is the one that, simultaneously

$$sensitivity = 100\% \wedge \max(AUC) \wedge \min(FPR) \tag{1}$$

We made the experiments using $sampleSize = 50,000$, $N = 50$ and W setted as a decreasing sequence from 123 to 50, and from that point further a value equal to 1. This gives more importance for the first ranking positions and less from the last ones. The FR algorithms used in the ensemble are: Information gain [6], Gain Ratio, Symmetrical Uncertainty [37] and Chi-square [25].

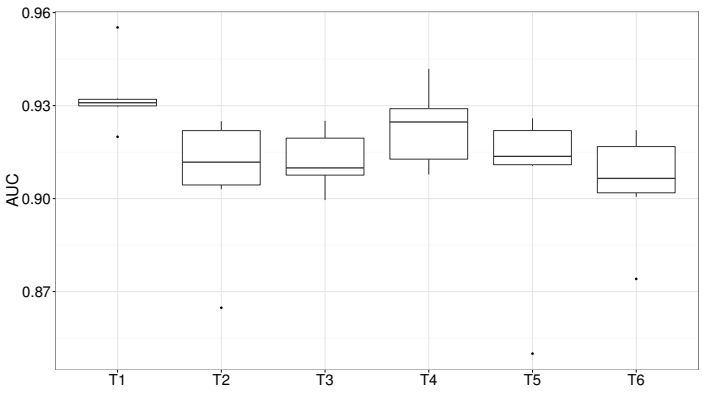

Fig. 3. AUC for different dataset constructions.

[2] http://www.r-project.org

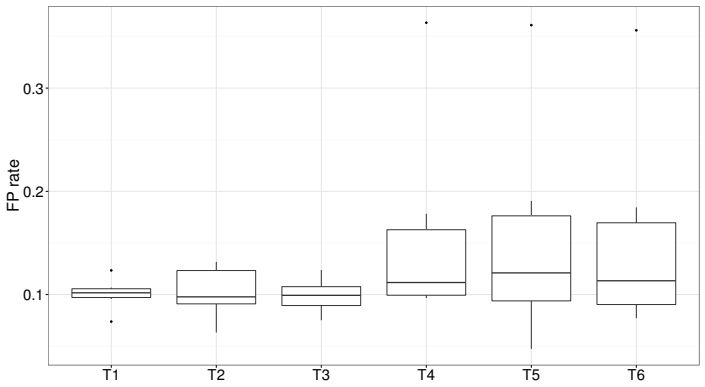

Fig. 4. FP rate for different dataset constructions.

In general, the proposed method for FR is the best, or among the best for producing classification models for the given dataset. Figure 3 shows the AUC variation in each of the dataset types, for all algorithms. As we can see, the datasets with FR (T_1 e T_4) produces better AUCs. T_1 gets an overall better performance. Besides, the NB and SVM seems to benefit most with the selection of features, using the proposed approach. Nevertheless, for the FPR, T_1 gives small variance in performance, but it is with T_5 that we get the best FPR value (see Figure 4). Relative to sensitivity, the best results are obtained for T_3, followed by T_1, as shown in Figure 5. Do notice the two best models, using the constraint in (1) are achieved with algorithm 1. Even for the RF algorithm, that includes its own feature ranking procedure, our FR approach achieved slightly better results in terms of AUC, as we can see in table 2.

5. Conclusions

This paper has given an account of and the reasons for the usage of FS techniques in classification problems. Prior studies have also noted the importance of FS as an important tool to improve classification. The purpose of the current study was to report an ensemble FS that can be used in large datasets, that are intractable as is, by some available FS procedures. This study has shown the usefulness our approach, towards the development of better classification models. We use a set of classification algorithms that covers the state-or-the-art learning schemes, evaluated using AUC, sensitivity and FPR. Based on our results, the AUC appears to be one of the best ways to evaluate a classifier performance. The following conclusions can be drawn from the present study. Our approach enables the usage of

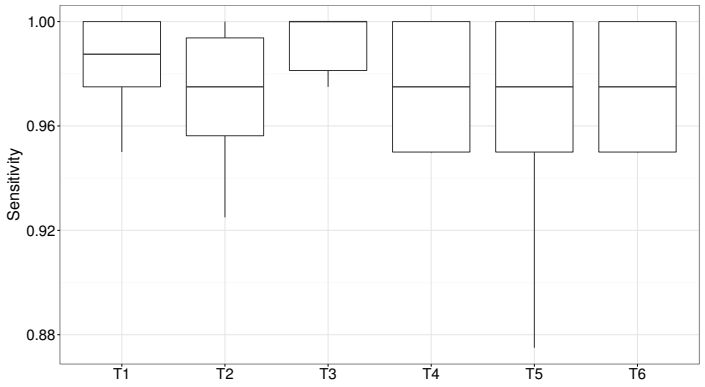

Fig. 5. Pacient sensitivity for different dataset constructions.

Table 2. Best combinations for maximising AUC and sensitivity.

Algo+Dataset	AUC	Patient *sensitivity*	FPR
NB+T1	0.955	100%	0.074
SVM L+T1	0.932	100%	0.102
SVM L+T2	0.925	100%	0.097
SVM P+T1	0.931	100%	0.106
SVM P+T3	0.912	100%	0.075
RF+T1	0.931	95%	0.101
RF+T3	0.925	97.5%	0.100
RF+T6	0.920	95%	0.086
SVM R+T1	0.930	97.5%	0.100
SVM R+T5	0.850	87.5%	0.047
BAG+T1	0.920	97.5%	0.123
BAG+T3	0.908	100%	0.124
BAG+T6	0.900	95%	0.102

known FR in large datasets. The result lead to the best results in AUC and FPR, in the tested dataset. We find that NB achieved the best performance with higher AUC and lower FPR. The evidence from this study suggests that FR is able to reduce the correlated variables, on many cases, responsible for the NB poor performance.

The current investigation was limited to one dataset. Our idea is to expand our testing, applying the same methodology in other datasets, to confirm the robustness of our FS method. Further work needs to be done to settle some parameters by default, namely, the size of the samples and the weights for the ranking procedure.

References

[1] Mehmet Fatih Akay. Support vector machines combined with feature selection for breast cancer diagnosis. *Expert Systems with Applications*, 36(2):3240–3247, 2009.

[2] Isaac Bankman. *Handbook of medical image processing and analysis*. Access Online via Elsevier, 2008.

[3] Michael Berry and Gordon Linoff. *Mastering Data Mining: The Art and Science of Customer Relationship Management*. John Wiley & Sons, Inc., New York, NY, USA, 1st edition, 1999.

[4] Michael J Berry and Gordon S Linoff. *Data mining techniques: for marketing, sales, and customer relationship management*. Wiley. com, 2004.

[5] Ella Bingham and Heikki Mannila. Random projection in dimensionality reduction: applications to image and text data. In *Proceedings of the seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 245–250. ACM, 2001.

[6] Kenneth P Burnham and David R Anderson. *Model selection and multi-model inference: a practical information-theoretic approach*. Springer Verlag, 2002.

[7] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.

[8] Tsang-Hsiang Cheng, Chih-Ping Wei, and Vincent S Tseng. Feature selection for medical data mining: comparisons of expert judgment and automatic approaches. In *Computer-Based Medical Systems, 2006. CBMS 2006. 19th IEEE International Symposium on*, pages 165–170. IEEE, 2006.

[9] Manoranjan Dash, Kiseok Choi, Peter Scheuermann, and Huan Liu. Feature selection for clustering-a filter solution. In *Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on*, pages 115–122. IEEE, 2002.

[10] Manoranjan Dash and Huan Liu. Feature selection for classification. *Intelligent Data Analysis*, 1(3):131–156, 1997.

[11] Ritendra Datta, Dhiraj Joshi, Jia Li, and James Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Comput. Surv.*, 40(2):5:1–5:60, May 2008.

[12] Michał Dramiński, Alvaro Rada-Iglesias, Stefan Enroth, Claes Wadelius, Jacek Koronacki, and Jan Komorowski. Monte carlo feature selection for supervised classification. *Bioinformatics*, 24(1):110–117, 2008.

[13] Chris Drummond, Robert C Holte, et al. C4.5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In *Workshop on Learning from Imbalanced Datasets II*, volume 11. Citeseer, 2003.

[14] Terrence S. Furey, Nello Cristianini, Nigel Duffy, David W. Bednarski, Michl Schummer, and David Haussler. Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, 16(10):906–914, 2000.

[15] Robert L Grossman. *Data mining for scientific and engineering applications*, volume 2. Springer, 2001.

[16] Isabelle Guyon. *Feature extraction: foundations and applications*, volume 207. Springer, 2006.

[17] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *J. Mach. Learn. Res.*, 3:1157–1182, March 2003.

[18] Mark A Hall. *Correlation-based feature selection for machine learning*. PhD thesis, The University of Waikato, 1999.

[19] Mark Andrew Hall and Geoffrey Holmes. Benchmarking attribute selection techniques for discrete class data mining. *Knowledge and Data Engineering, IEEE Transactions on*, 15(6):1437–1447, 2003.
[20] George H John, Ron Kohavi, Karl Pfleger, et al. Irrelevant features and the subset selection problem. In *ICML*, volume 94, pages 121–129, 1994.
[21] Ron Kohavi and George H John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1):273–324, 1997.
[22] Igor Kononenko. Estimating attributes: analysis and extensions of RELIEF. In *Machine Learning: ECML-94*, pages 171–182. Springer, 1994.
[23] Pat Langley and Stephanie Sage. Induction of selective bayesian classifiers. In *Proceedings of the Tenth international conference on Uncertainty in artificial intelligence*, pages 399–406. Morgan Kaufmann Publishers Inc., 1994.
[24] Patrick Langley and Stephane Sage. Scaling to domains with irrelevant features. *Computational Learning Theory and Natural Learning Systems*, 4:51–63, 1997.
[25] Albert M Liebetrau. *Measures of association*, volume 32. Sage Publications, Incorporated, 1983.
[26] Huan Liu and Hiroshi Motoda. *Feature selection for knowledge discovery and data mining*. Springer, 1998.
[27] Tatsunori Mori. Information gain ratio as term weight: the case of summarization of IR results. In *Proceedings of the 19th International Conference on Computational Linguistics - Volume 1*, COLING '02, pages 1–7, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.
[28] J. Ross Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986.
[29] John Ross Quinlan. *C4. 5: programs for machine learning*, volume 1. Morgan Kaufmann, 1993.
[30] Yong Rui, Thomas S Huang, and Shih-Fu Chang. Image retrieval: Current techniques, promising directions, and open issues. *Journal of Visual Communication and Image Representation*, 10(1):39–62, 1999.
[31] Chris Rygielski, Jyun-Cheng Wang, and David C. Yen. Data mining techniques for customer relationship management. *Technology in Society*, 24(4):483 – 502, 2002.
[32] Yvan Saeys, Iaki Inza, and Pedro Larraaga. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23(19):2507–2517, 2007.
[33] Vítor Santos, Nuno Datia, and M.P.M. Pato. Classification performance of data mining algorithms applied to breast cancer data. In Jo ao Manuel R.S. Tavares and R.M. Natal Jorge, editors, *Proceedings of the IV ECCOMAS Thematic Conference on Computational Vision and Medical Image Processing: VipIMAGE 2013*. Taylor and Francis, October 2013.
[34] A. Sharma, C. H. Koh, S. Imoto, and S. Miyano. Strategy of finding optimal number of features on gene expression data. *Electronics Letters*, 47(8):480–482, 2011.
[35] Blake Shaw and Tony Jebara. Structure preserving embedding. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 937–944. ACM, 2009.
[36] Simon Tong and Edward Chang. Support vector machine active learning for image retrieval. In *Proceedings of the ninth ACM international conference on Multimedia*, pages 107–118. ACM, 2001.
[37] Ian H Witten and Eibe Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2005.
[38] Lei Yu and Huan Liu. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *ICML*, volume 3, pages 856–863, 2003.